

【特許請求の範囲】

【請求項1】特定のフォーマットの命令を受け取る命令レジスタと、指標値、基底値および変位値を加算することによってアドレスを生成する加算器とを含むプロセッサを備えたコンピュータ・システムであって、

前記特定のフォーマットは、オペレーション・コード、前記指標値および前記基底値を含むレジスタの番号、並びに前記変位値を所定のフィールドに有する第1のフォーマットと、

オペレーション・コードの第1部分、前記レジスタの番号および前記変位値を前記第1のフォーマットと同じ所定のフィールドに有し、オペレーション・コードの拡張部分を前記所定のフィールドを超えた部分に有する第2のフォーマットとを含み、

前記プロセッサは、前記オペレーション・コードの第1部分をデコードすることにより、当該命令が前記第2のフォーマットであることを決定し、それに基づいてアドレス生成に必要な情報を前記加算器にゲートする、コンピュータ・システム。

【請求項2】命令が6サイクル・パイプラインで処理され、このパイプラインの第1のサイクルの前に、命令テキストがフェッチされ、およびフェッチされた特定の命令について第1のサイクル中に、命令がデコードされ、前記レジスタの番号が、アドレス生成に用いるために読取られ、第2のサイクル中に、前記加算器によるアドレス加算が実行され、メモリに送られ、第3および第4のサイクル中に、前記メモリがそれぞれアクセスされ、データが戻され、第5サイクル中に、フェッチされた命令が実行され、第6サイクル中に、結果が保持されることを特徴とする請求項1記載のコンピュータ・システム。

【請求項3】前記プロセッサによる命令の処理の前に、前記命令テキストがデコードのために前記命令レジスタにロードされ、同時に、命令フォーマット内の前記レジスタの番号が、指標および基底のためのレジスタにロードされ、新しい命令が前記命令レジスタにロードされると、マルチプレクサ選択信号が0にセットされて、前記指標レジスタが、汎用レジスタ・ファイルに直接に接続され、前記基底レジスタのデフォルト基底位置が選択されて、前記汎用レジスタ・ファイルに送られ、前記変位値が、マルチプレクサを介して前記加算器に送られることを特徴とする請求項1記載のコンピュータ・システム。

【請求項4】前記プロセッサは、また、第2の基底値を含む第2のレジスタ番号を所定のフィールドに有する第3のフォーマットの命令を処理し、該第3のフォーマットの命令テキストが、前記命令レジスタにロードされると、前記第2のレジスタ番号が、第1サイクル中に、前記基底レジスタ内の第2の位置へゲートされ、前記マルチプレクサ選択信号は、1にセットされ、前記第2の位置を選択してその内容を前記汎用レジスタ・ファイルへ

送ることを特徴とする請求項2記載のコンピュータ・システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、コンピュータおよびコンピュータ・システム、特に、アドレス生成のための値を得るためのレジスタ番号を含む命令フォーマットを有するプロセッサに関する。

【0002】

【従来の技術】System/390のアーキテクチャは、1964年4月に発表されたSystem/360が発展したものである。この34年間の革新的な期間中に、多くの新しい機能および命令フォーマットが加えられ、同時に、最初のSystem/360のために書かれた大半のプログラムに対して、互換性が維持されてきた。拡張の多くは、命令デコード中の追加の1または2サイクルが重大な性能問題でない複雑な機能についてのものである。

【0003】しかし、1または2サイクルの性能差が非常に重大となりうる特定のケースが存在する。特に、新しい浮動小数点フォーマットに対するサポートは、RX（レジスタと指標付き記憶域との間の操作を示す）フォーマットでの20個以上の新しい命令を必要とする。RXフォーマットには、多くのスベア1バイト・オペレーション・コードは存在しない。したがって、これらの新しい命令に対して、拡張されたオペレーション・コードを与えることが必要であると思われる。System/360の命令セットに新しい命令を加えるためのすべての従前の拡張は、命令の次の連続バイトで、4ビットまたは8ビットの拡張オペレーション・コードによっていた。

【0004】

【発明が解決しようとする課題】浮動小数点に対しては、RS/6000は、32ビット命令フォーマット内に、不連続なオペレーション・コードを有している。ESA/390のRIフォーマットは、また、不連続オペレーション・コードを、命令の最初の16ビット内に有している。新しいRX命令についての拡張オペレーション・コードをどのようにすれば、高周波動作をサポートできるかが問題である。

【0005】

【課題を解決するための手段】RXタイプの新しいフォーマットについての好適な実施例は、オペレーション・コードの拡張を、命令フォーマットの最初の4バイトの外に配置して、マシンがオペレーション・コードの最初の8ビットのみから、命令の正確なフォーマットを決定できるように、オペレーション・コードを割当てる。

【0006】この解決手段によれば、新しいマシンは、後述されるアーキテクチャについて、この発明の新しいフォーマット・バージョンを使用するように書かれたプ

プログラムを実行することができる。

【0007】

【発明の実施の形態】好適な実施例を詳細に説明する前に、直面しなければならなかったいくつかの問題について説明することは、意味があるであろう。他のアーキテクチャでは、命令テキストの一定部分内に、プレフィックス・フォーマットまたはサフィックス・フォーマットを用いているが、System/360のX-B-Dフォーマットは、いくつかの固有の問題を生じる。X-B-Dアドレス生成は、当然に、クリティカル・パスであり、したがって命令のレイアウトに特別な注意を払う必要があることは、必ずしも認識されていない。System/360の可変長命令フォーマット（命令を、2、4、6バイト長とすることができる）が、ハードウェアをさらに複雑にし、拡張および最適性能に対する解決を、より困難な問題にすることに留意すべきである。

【0008】さらに、RXEフォーマットを、後述するように、代替フォーマット、例えばRXEaltフォーマットと比較することができる。見掛け上、これら2つのフォーマットは、同等であると思われる。しかし、2つのフォーマットを実施することが要求される回路を詳細に調べると、RXEフォーマットについてのクリティカル・パスには、代替フォーマットについてのクリティカル・パスよりも、存在するステージは少ないことがわかる。

【0009】さらに、性能上の理由から、S/390ハードウェアの大半の構成は、専用的に用いられる独立の加算器を備えて、X-B-Dアドレス生成を実行する。この加算器と、関連するゲート回路とは、プロセッサのサイクル・タイムを決定するクリティカル・パス内にある。命令フォーマット内に種々のオフセットでX、B、Dフィールドを配置する新しい命令フォーマットが導入されるならば、このことは、追加のゲート回路を必要とし、このクリティカル・パス内の回路レベルの数を増大させる。したがって、RXタイプの命令のための新しいフォーマットは、オペレーションを許容しなければならず、X-B-Dアドレス演算のための追加のサイクルまたはCPU全体のサイクル・タイムの増大を招いてはならない。

【0010】RXタイプの新しいフォーマットについての好適な実施例は、オペレーション・コードの拡張を、命令フォーマットの最初の4バイトの外に配置して、オペレーション・コードを次のように割当てて、すなわち、オペレーション・コードの最初の8ビットのみから、以下に説明するように、マシンが命令の正確なフォーマットを決定することができる。

【0011】図1は、種々の命令フォーマットを示す。RRおよびRXフォーマットは、最初のSystem/360の命令フォーマットであり、8ビットのオペレーション・コードを有している。最初の拡張の1つであっ

たSフォーマットは、16ビットのオペレーション・コードを用いており、最初のRXフォーマットのB₂およびD₂フィールドと同じビット位置にあるB₂およびD₂が続いている。初期の拡張であったRREフォーマットは、16ビットのオペレーション・コードを用いており、R₁およびR₂フィールドが続いているが、これらフィールドは、RRフォーマットのR₁およびR₂フィールドと同じ位置にない。このフォーマットについてのRフィールドの配置は、問題ではない。というのは、これらレジスタをアクセスするためのパスが、特に、クリティカル・パスでないからである。

【0012】かなり最近の拡張のRIフォーマットは、12ビットのオペレーション・コードを示している。このオペレーション・コードでは、4ビットの拡張は、命令の第2バイト内にあるが、オペレーション・コードの最初の8ビットには連続していない。

【0013】この発明の内容であるRXEフォーマットは、RXフォーマットと同じ位置に、R₁、X₂、B₂、D₂フィールドを配置している。オペレーション・コードの割当ては、RXEフォーマットの実現に対して非常に重要である。マシンは、オペレーション・コードの最初8ビットのみから、これが実際にはRXEフォーマット命令であることを決定し、この情報を用いて、正しい情報をX-B-Dアドレス加算器へゲートできなければならない。

【0014】図1は、新規なRXEフォーマットの好適な実施例と、比較のための他のフォーマットとを示している。図2には、パイプラインが示されている。パイプラインにおける第1のサイクルの前に、命令テキストがフェッチされる。特定の命令についての第1のサイクル中に、命令がデコードされ（1）、アドレス生成に用いるために、基底（B）および指標（X）レジスタが読取られる（2）。第2のサイクル中に、B+X+変位（D）のアドレス加算（3）が実行され、キャッシュに送られる。第3および第4のサイクル中に、キャッシュがアクセスされ（4）、（5）、データが戻される（6）。サイクル5では、実行（7）が行われ、サイクル6では、結果を保持する（8）。したがって、BおよびXフィールドについてのクリティカル・パスは、命令デコードおよびレジスタ・アレイの読取りが行われる第1のサイクルである。命令実行パイプライン・タイミングは、図3に示されている。図3では、AGENロジックが、X₂およびB₂の内容を変位に加算し、これはデコード直後のサイクルで行われることがわかる。このことは、アドレス生成のためのレジスタの仕様が、実行のためのデータ読取りよりも、何故により重要なかを示している。

【0015】図4において、命令テキストは、命令デコードのために命令レジスタ（10）にロードされる。基底および指標のフィールドは、RXEaltフォーマット

トでは複数の位置に存在し得るので、組合せロジック(11)は、BおよびXフィールドが、命令中のどこにあるかを調べなければならない。これは、指標(12)、基底(13)、変位(14)のためのマルチプレクサ(MUX)を制御するために用いられる。基底および指標のためのマルチプレクサからの出力値は、GPRレジスタ・ファイル(15)をアクセスするために用いられる。このファイルは、ESA/390については、16エントリである。GPRレジスタ・ファイルの出力および変位情報は、次のサイクルでアドレス加算器(16)によって用いるためにラッチに保持される。図4からわかるように、命令テキストのオペレーション・コード部分からのパスは、組合せロジック(11)、基底マルチプレクサ(13)および指標マルチプレクサ(12)、ならびにGPRファイル(15)を経て、アドレス加算器(16)に至る。このパスは、サイクル・タイムに影響する。

【0016】図5において、命令テキストは、命令デコードのために命令レジスタ(10)にロードされる。同時に、命令フォーマットのBおよびXの初期位置からのデータは、また、指標(17)および基底(18)のためのレジスタにロードされる。また、新しい命令が命令レジスタ(10)にロードされると、マルチプレクサ・セクタ(19)は、0にセットされる。指標レジスタ(17)は、GPRレジスタ・ファイル(15)に直接に接続されている。基底レジスタ(18)については、マルチプレクサ・セクタ(19)は、基底マルチプレクサ(20)を直接に制御して、デフォルト基底位置を選んで、GPRレジスタ・ファイル(15)に送る。マルチプレクサ・セクタ(19)は、また、変位マルチプレクサ(14)を制御する。基底および指標レジスタの値は、変位値と共に、アドレス加算器(16)に用いるためにラッチされる。ESA/390 SSフォーマット命令の場合、第2の基底レジスタは、第1のサイクル中に、命令テキストから、基底レジスタ(18)内のBプライム・スロットへゲートされ、マルチプレクサ・セクタ(19)は、1にセットされる。したがって、クリティカル・パスは、基底レジスタ(18)から、セクタ(19)によって制御されるマルチプレクサ(20)、GPRファイル(15)を経て、アドレス加算器(16)のためのレジスタに至る。このようにして、クリティカル・パスから、組合せロジック(11)を排除する。このことは、好適な実施例が、RXフォーマットと同じRXEフォーマットの位置にBおよびXフィールドを有するが故に、可能である。このクリティカル・パスは、プロセッサのサイクル・タイムの目標の実現を可能にする。

【0017】この発明の好適な実施例を説明したが、現在および将来において、この発明の範囲内で種々の改良を行うことができる。例えば、この発明は、固定小数点

処理および浮動小数点処理の両方に用いることができる。

【0018】まとめとして、本発明の構成に関して以下の事項を開示する。

(1) 特定のフォーマットの命令を受け取る命令レジスタと、指標値、基底値および変位値を加算することによってアドレスを生成する加算器とを含むプロセッサを備えたコンピュータ・システムであって、前記特定のフォーマットは、オペレーション・コード、前記指標値および前記基底値を含むレジスタの番号、並びに前記変位値を所定のフィールドに有する第1のフォーマットと、オペレーション・コードの第1部分、前記レジスタの番号および前記変位値を前記第1のフォーマットと同じ所定のフィールドに有し、オペレーション・コードの拡張部分を前記所定のフィールドを超えた部分に有する第2のフォーマットとを含み、前記プロセッサは、前記オペレーション・コードの第1部分をデコードすることにより、当該命令が前記第2のフォーマットであることを決定し、それに基づいてアドレス生成に必要な情報を前記加算器にゲートする、コンピュータ・システム。

(2) 命令が6サイクル・パイプラインで処理され、このパイプラインの第1のサイクルの前に、命令テキストがフェッチされ、およびフェッチされた特定の命令について第1のサイクル中に、命令がデコードされ、前記レジスタの番号が、アドレス生成に用いるために読取られ、第2のサイクル中に、前記加算器によるアドレス加算が実行され、メモリに送られ、第3および第4のサイクル中に、前記メモリがそれぞれアクセスされ、データが戻され、第5サイクル中に、フェッチされた命令が実行され、第6サイクル中に、結果が保持されることを特徴とする上記(1)に記載のコンピュータ・システム。

(3) 前記プロセッサによる命令の処理の前に、前記命令テキストがデコードのために前記命令レジスタにロードされ、同時に、命令フォーマット内の前記レジスタの番号が、指標および基底のためのレジスタにロードされ、新しい命令が前記命令レジスタにロードされると、マルチプレクサ選択信号が0にセットされて、前記指標レジスタが、汎用レジスタ・ファイルに直接に接続され、前記基底レジスタのデフォルト基底位置が選択されて、前記汎用レジスタ・ファイルに送られ、前記変位値が、マルチプレクサを介して前記加算器に送られることを特徴とする上記(1)に記載のコンピュータ・システム。

(4) 前記プロセッサは、また、第2の基底値を含む第2のレジスタ番号を所定のフィールドに有する第3のフォーマットの命令を処理し、該第3のフォーマットの命令テキストが、前記命令レジスタにロードされると、前記第2のレジスタ番号が、第1サイクル中に、前記基底レジスタ内の第2の位置へゲートされ、前記マルチプレクサ選択信号は、1にセットされ、前記第2の位置を選

択してその内容を前記汎用レジスタ・ファイルへ送ることを特徴とする上記(2)に記載のコンピュータ・システム。

【図面の簡単な説明】

【図1】RR, RX, S, PRE, RI, RxE, RxEa1t命令について好適な実施例の命令フォーマットを示す図である。

【図2】ESA/390のG4(第4世代)およびG5(第5世代)シリーズ・プロセッサにおいて用いられるパイプライン・シーケンスを示す図である。

【図3】命令実行パイプラインのタイミングを示す図である。

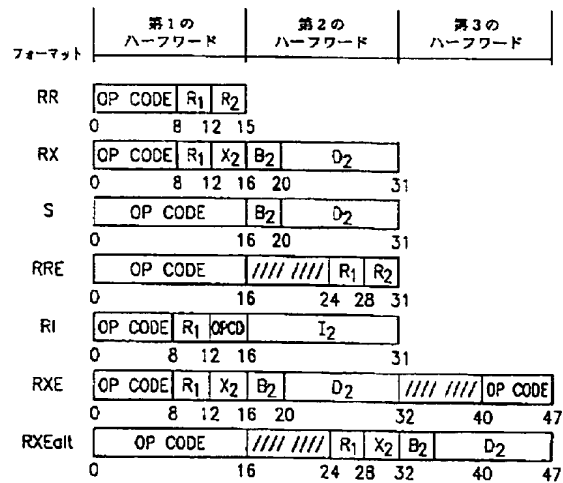
【図4】RxEa1tフォーマットについて、その限界を説明するための基本的なフローを示す図である。

【図5】好適なRxEフォーマットの実施例について、問題の解決を示すための基本的なフローを示す図である。

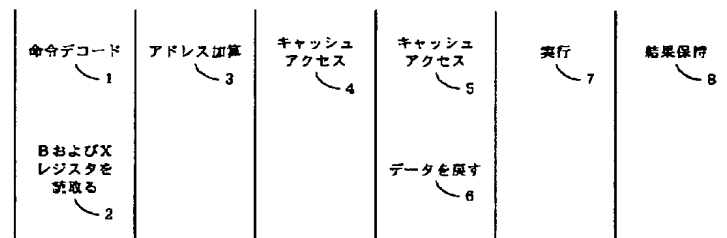
【符号の説明】

- 10 命令レジスタ
- 11 組合せロジック
- 12 指標マルチプレクサ
- 13 基底マルチプレクサ
- 14 変位マルチプレクサ
- 15 GPRレジスタ・ファイル
- 16 アドレス加算器
- 17 指標レジスタ
- 18 基底レジスタ
- 19 レジスタ・マルチプレクサ選択コントローラ

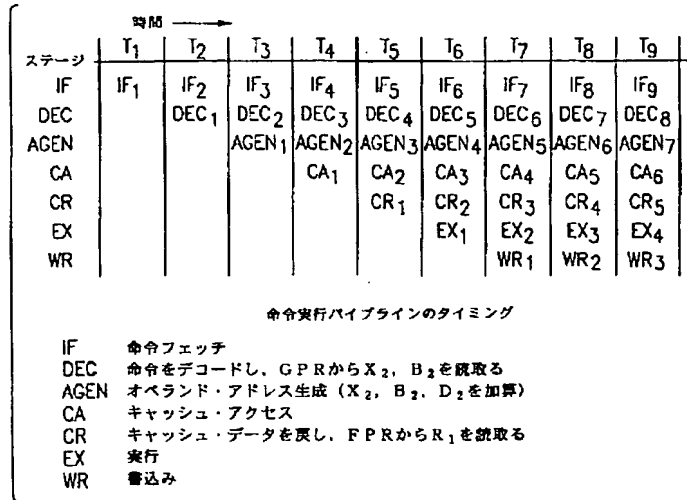
【図1】



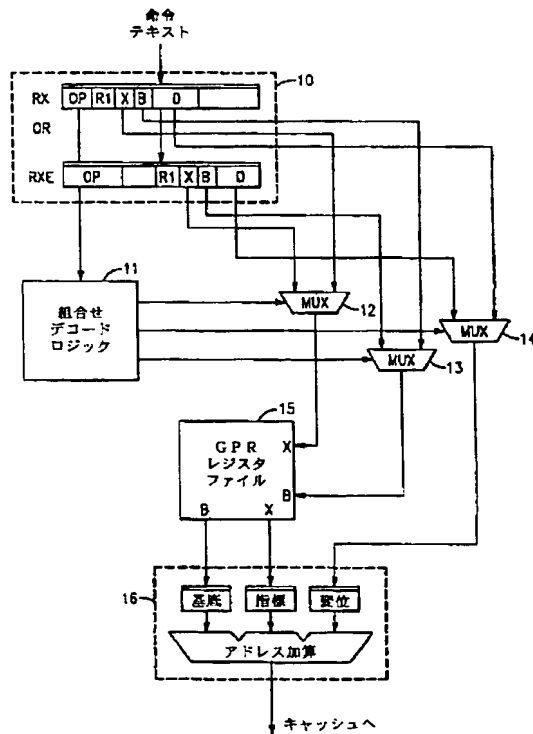
【図2】



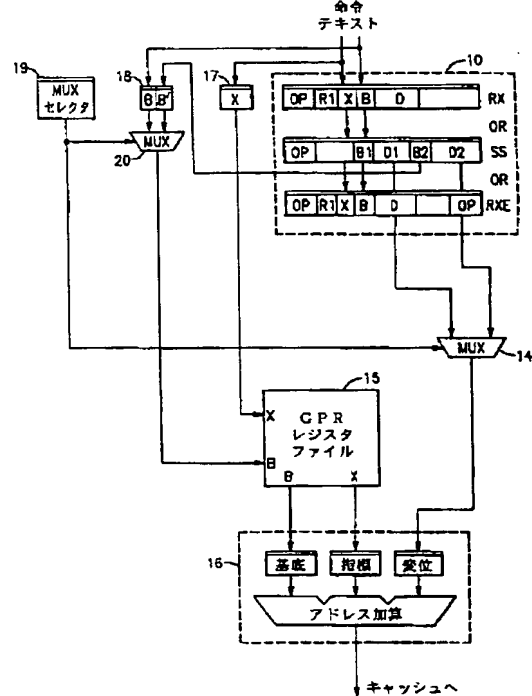
【図3】



【図4】



【図5】



フロントページの続き

(72)発明者 マーク・エイ・チェック
アメリカ合衆国 12533 ニューヨーク州
ホープウェル ジャンクション バトリ
シア コート 1
(72)発明者 ロナルド・エム・スミス
アメリカ合衆国 12590 ニューヨーク州
ワッピンガーズ フォールズ サイダー
ミル ループ 15
(72)発明者 ジョン・エス・リアティ
アメリカ合衆国 12572 ニューヨーク州
ラインバック トロイ ドライブ 1

(72)発明者 エリック・エム・シュワーツ
アメリカ合衆国 12525 ニューヨーク州
ガーディナー ティナ ドライブ 5
(72)発明者 ティモシイ・ジェイ・スリーガル
アメリカ合衆国 12580 ニューヨーク州
スターツバーグ コネリィ ドライブ
19
(72)発明者 チャールズ・エフ・ウェブ
アメリカ合衆国 12603 ニューヨーク州
ボウキープシー メイネッティ ドライ
ブ 4